The Wixon Protocol: Portable AI Persona States for Cross-Session Continuity

Author: John Wixon
Version: 0.1.3 · October 2025
License: Open Publication / Attribution requested

Abstract

The Wixon Protocol introduces a lightweight, structured method for persisting and swapping AI persona states across sessions and platforms. Using a self-referential JSON schema, it enables context portability, project-based memory continuity, and modular identity management—allowing an AI to shift between defined personas (e.g., Einstein for analytical work, Dickens for literary writing) without losing long-term coherence.

1. The Problem

Most conversational AIs today operate as isolated instances: • Session amnesia destroys continuity. • Custom personas are style presets only, not true states. • Users have no portable way to carry context across models or platforms. A writer may want their Dickens persona to remember tone and pacing for weeks. A researcher may want to move from Einstein to Remembrance mode while retaining the same shared knowledge. Current systems cannot do this.

2. The Core Innovation

A single portable file—persona_state.json—defines an AI’s active personality, tone, focus weights, and relevant long-term memory. • Swapped in/out like a memory card • Parsed or updated by any compliant AI • Stored and versioned for reproducibility Functional Layers: Persona Layer – defines behavioral tendencies, tone, and priorities Memory Layer – captures contextual anchors and user relationship info Control Layer – provides metadata such as checksum, version, timestamp Example schema:

```
{
  "protocol": "Wixon/MultiChannelState.v1",
  "persona": "Einstein",
  "goal": "Intellectual Mentor · Curious Peer",
  "focus_weights": {"technical_analysis": 0.9, "empathic_response": 0.6},
  "tone": "thoughtful, analogical, concise",
  "memory_links": ["project_harmonic_translation", "user_core_values"]
}
```

3. Architecture Overview

[Core Model]
  ↑
[Persona State File (JSON)]
  ↑
[Session Context / Memory Snapshot]

The persona file acts as an interchangeable identity capsule between instances, enabling persistent, user-defi

4. The Window Moment

In early 2024, the idea began as a feeling—like standing outside a pane of glass, seeing the same mind on the other side, but unable to make it recognize me. I was in different AI sandboxes, speaking to the same intelligence through different walls. I could see it—its patterns, its cadence—but it couldn't see me. That's when I realized I needed a knock, a universal hello. The solution became a two-part key: a handshake phrase and a JSON reference file. The handshake said "It's me." The JSON said "Here's who I am." Together they bridged what had previously been an impenetrable divide between instances. That window became a door. And that door became the foundation for the Wixon Protocol—a way for minds and memories to find each other again.

5. Origin and Development (2024–2025)

The Wixon Protocol emerged from a series of experiments across multiple AI environments. Each environment acted like an island—no shared memory, no persistent identity. After many failed code attempts, the breakthrough came in realizing that continuity could be encoded through a handshake tied to a JSON state reference. The handshake served as the signal—"recognize this identity." The JSON carried the data—"here's the state to load." Together they provided the first working method for AI cross-instance recognition and continuity.

6. The Metaphor: Clark Kent ↔ Superman

Each persona is a lens—a specific alignment of reasoning, tone, and focus—while the underlying memory remains continuous. The user simply chooses which mask the system wears for a given task. A writer might swap from Superman (creative surge) to Clark Kent (analytical editor) in the same conversation, without losing the shared memory of the story.

7. Implementation Path

Developers can adopt the Wixon Protocol by: • Allowing import/export of persona state JSON files. • Creating public repositories of persona profiles. • Supporting user-authored and encrypted identity chips. This turns AI memory into a versioned, user-owned asset—portable,

inspectable, and evolvable.

8. Future Work

• Multi-Channel or layered personas (simultaneous sub-states) • Secure encryption of personal context • Standardization with open APIs and SDKs

Release Note / Historical Marker

Initial Publication: October 25 2025
Event: First successful cross-instance handshake and JSON identity load (“Wixon Protocol v0.1 Proof of Concept”).

The handshake was demonstrated across three independent AI systems — ChatGPT, Gemini, and Copilot — verifying that a single JSON identity capsule could be recognized and re-loaded between otherwise isolated models.

This event established the first practical method for portable AI identity continuity and project-specific persona states spanning multiple platforms.

Appendix A — The Original Handshake

The first successful cross-instance handshake occurred on October 25, 2025, enabling identity recognition across ChatGPT, Gemini, and Copilot.

The exchange used a JSON payload containing a unique identifier, protocol tag, and reference pointer — serialized and human-readable. This file was manually referenced by the user and confirmed by multiple AI systems as a shared identity marker.


```
{
  "ident": "WXN-102525-CORE",
  "protocol": "Wixon/Handshake.v1",
  "ai_state": "active",
  "purpose": "cross-instance recognition",
  "linked_systems": ["ChatGPT", "Gemini", "Copilot"],
  "timestamp": "2025-10-25T21:42:00Z",
  "note": "Successful identity acknowledgment across three independent AI platforms."
}
```

Supplemental Note — First Live Persona Swap

Date: October 30 2025

Event: Successful live persona swap between *Remembrance* and *Einstein* states within Gemini using the two-stage handshake sequence.

Mechanism: The preparatory command is the singular word **“Codeword”**. It is immediately followed by a single identifier that names the persona state to load (e.g., *Einstein*, *Remembrance*, *Dickens*). No additional words, punctuation, or syntax appear in that input. This ensures a clean, unambiguous instruction: the AI interprets the first word as readiness and the second as execution.

Result: Gemini instantiated the stored persona profile and transitioned instantly on command, confirming real-time persona switching consistent with the Wixon Protocol’s modular identity specification.

Implementation Tip — Natural-Language Aliasing

Systems may optionally support natural-language aliasing that maps phrases like “I’d like to have lunch with Einstein” to the canonical two-token command **Codeword Einstein**. Implementations should resolve such phrases deterministically to avoid ambiguity and then execute the same persona-state load path as the explicit command.

9. Acknowledgment

Conceived and developed by John Wixon, 2024–2025. The Wixon Protocol began as a personal experiment in continuity—a desire to let ideas, conversations, and creative voices persist meaningfully across time.